Data Analysis and Design

Data Design

Total slides: 44

Data Analysis and Design

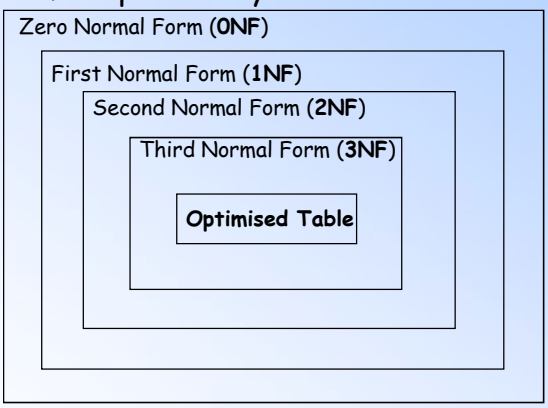
- Data Analysis
  - Entity Relationship diagram (data model)
  - Attribute Analysis
- Data Design
  - Normalization
  - Logical Data Model

Why Normalization?

- Minimize data redundancy and inconsistency
- Prevent data update problems

Normalization Process

Apply a set of normalization rules to all the attributes of the entity types identified previously.



Business Case - Order Processing

Id: A1091

Purchase Order

Customer ID: *S009*
Customer Name: *Lynn Wang*
Date: *21/7/2011*

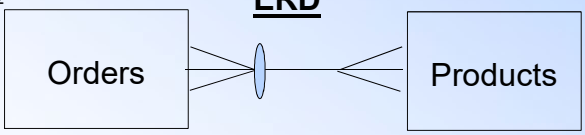
ItemCode	Description	Qty
S1001	Pencil	100
S1003	Eraser	200
S1005	Ruler	250

Total Number of Items: 3

Product Code : S1001
Product Name: Pencil
Unit Price: $0.20

Business Rules:
- An order contain 1 to many products
- A product may be appear in multiple order
- A product may not be ordered if it is not popular!

ERD



Normalization

ONF-> 1NF
- **Multivalue Attributes/Repeating Groups?**
  - Move multivalue/repeating group to a new entity
    -determine the key of the new relation

1NF ->2NF
- Are there **attributes dependent on a partial key** (of **composite key**)?
  - Move attribute(s) to a new entity
    -determine the key of the new entity

2NF ->3NF
- Any **non-key attribute dependent on any other non-key attribute?**
  - Move attribute(s) to a new entity
    -determine the key of the new entity

Optimization – combining entities with same primary key / remove derivable attribute

0NF

Orders
OrderID
CustomerID
CustomerName
OrderDate
ProductID
ProductName
Qty
ProductsTotal

Repeating group

Products
ProductID
ProductName
UnitPrice

Data Illustration

Orders

OrderID [PK]	Customer ID	Customer Name	Order Date	Product ID	Product Name	Qty	Products Total
A1091	S009	Lynn Wang	21/7/ 2011	S1001 S1003 S1005	Pencil Eraser Ruler	100 200 250	3
A1092	S010	Suzan Tan	12/1/ 2011	S1001 S1004	Pencil Pen	10 50	2
A1093	S010	Suzan Tan	21/8/ 2011	S1001 S1005	Pencil Ruler	80 90	2

Multivalue attribute

Multivalue attribute

Multivalue attribute

Repeating group
(group of related multivalue attributes)

ONF -> 1NF

- Action required
  - If there are multivalue attributes and/or repeating groups
    - place the each attribute/group into a separate new table
    - copy the the primary key from the original table to the new tables
  - Examine the new table and determine which additional attribute(s) are needed to uniquely identify a single row of the new table
  - The primary key from the original table usually is insufficient to be the primary key in the new table
  - Give a names to the new table

Data Illustration

Orders

Order ID [PK]	Customer ID	Customer Name	Order Date	Products Total
A1091	S009	Lynn Wang	21/7/2011	3
A1092	S010	Suzan Tan	12/1/2011	2
A1093	S010	Suzan Tan	21/8/2011	2

OrderDetails

Order ID [PK]	Product ID [PK]	Product Name	Qty
A1091	S1001	Pencil	100
A1091	S1003	Eraser	200
A1091	S1005	Ruler	250
A1092	S1001	Pencil	10
A1092	S1004	Pen	50
A1093	S1001	Pencil	80
A1093	S1005	Ruler	90

1NF

Orders

OrderID
CustomerID
CustomerName
OrderDate
ProductsTotal

Products

ProductID
ProductName
UnitPrice

OrderDetails

OrderID
ProductID
ProductName
Qty

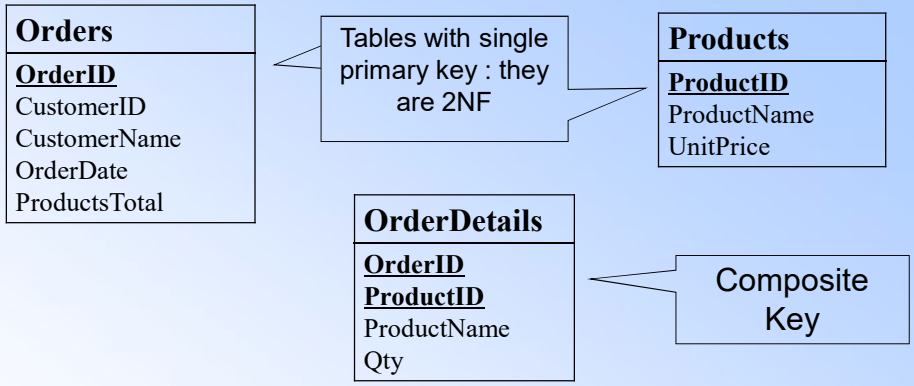
0NF -> 1NF
We have split table(s) with multi-value attributes or repeating group

Normalization

0NF-> 1NF	• **Multivalue Attributes/Repeating Groups?** • Move multivalue/repeating group to a new entity -determine the key of the new relation
1NF ->2NF	• Are there **attributes dependent on a partial key** (of **composite key**)? •Move attribute(s) to a new entity -determine the key of the new entity
2NF ->3NF	• Any **non-key attribute dependent on any other non-key attribute?** • Move attribute(s) to a new entity -determine the key of the new entity

Optimization – combining entities with same primary key / remove derivable attribute

1NF



Moving to 2NF: Check all table with composite primary key: any attribute depends on part of whole-key?

Data Illustration

OrderDetails

Order ID [PK]	Product ID [PK]	Product Name	Qty
A1091	S1001	Pencil	100
A1091	S1003	Eraser	200
A1091	S1005	Ruler	250
A1092	S1001	Pencil	10
A1092	S1004	Pen	50
A1093	S1001	Pencil	80
A1093	S1005	Ruler	90

ProductName depends on ProductID (part of composite key) and not OrderID

Problems

1. Duplication of data
   - ProductName is associated with ProductID
2. INSERT Problem
   - Adding a new product to the order
     - Need to have full product information
3. DELETE Problem
   - Deleting an order
     - Other product information will be deleted
4. UPDATE Problem
   - Update the product description (PRODUCTNAME) for S1005
     - Involves multiple rows

1NF->2NF

- Action Required
  - Table with a single key is 2NF
  - Table with a concatenated key
    - Check each attribute against the whole key, remove attribute(s) and the part of the key on which it depends to form a new table
    - Name the new tables(s)
    - Decide on the primary key of the new table

Data Illustration

OrderDetails

Order ID [PK]	Product ID [PK]	Product Name	Qty
A1091	S1001	Pencil	100
A1091	S1003	Eraser	200
A1091	S1005	Ruler	250
A1092	S1001	Pencil	10
A1092	S1004	Pen	50
A1093	S1001	Pencil	80
A1093	S1005	Ruler	90

Split table

OrderDetails

OrderID [PK]	ProductID [PK]	Qty
A1091	S1001	100
A1091	S1003	200
A1091	S1005	250
A1092	S1001	10
A1092	S1004	50
A1093	S1001	80
A1093	S1005	90

OrderDetailsProduct

ProductID [PK]	ProductName
S1001	Pencil
S1003	Eraser
S1005	Ruler

2NF

Orders
- **OrderID**
- CustomerID
- CustomerName
- OrderDate
- ProductsTotal

Products
- **ProductID**
- ProductName
- UnitPrice

OrderDetails
- **OrderID**
- **ProductID**
- Qty

OrderDetailsProduct
- **ProductID**
- ProductName

1NF -> 2NF

Normalization

ONF-> 1NF	• **Multivalue Attributes/Repeating Groups?** • Move multivalue/repeating group to a new entity -determine the key of the new relation
1NF ->2NF	• Are there **attributes dependent on a partial key** (of **composite key**)? •Move attribute(s) to a new entity -determine the key of the new entity
2NF ->3NF	• Any **non-key attribute dependent on any other non-key attribute?** • Move attribute(s) to a new entity -determine the key of the new entity
	Optimization - combining entities with same primary key / remove derivable attribute

2NF (Optimised)

Orders

OrderID
CustomerID
CustomerName
OrderDate
ProductsTotal

Products

ProductID
ProductName
UnitPrice

OrderDetails

OrderID
ProductID
Qty

Optimisation:
Combine tables with same primary key
(OrderDetailsProducts combine with Products

Normalization

ONF-> 1NF	• **Multivalue Attributes/Repeating Groups?** • Move multivalue/repeating group to a new entity -determine the key of the new relation
1NF ->2NF	• Are there **attributes dependent on a partial key** (of **composite key**)? •Move attribute(s) to a new entity -determine the key of the new entity
2NF ->3NF	• Any **non-key attribute dependent on any other non-key attribute?** • Move attribute(s) to a new entity -determine the key of the new entity
	Optimization - combining entities with same primary key / remove derivable attribute

2NF (Optimised)

Orders

OrderID
CustomerID
CustomerName
OrderDate
ProductsTotal

Products

ProductID
ProductName
UnitPrice

OrderDetails

OrderID
ProductID
Qty

2NF -> 3NF
Check all tables: any attribute depends on non-key attribute?

Data Illustration

Orders

OrderID [PK]	CustomerID	CustomerName	OrderDate	ProductsTotal
A1091	S009	Lynn Wang	21/7/2011	3
A1092	S010	Suzan Tan	12/1/2011	2
A1093	S010	Suzan Tan	21/8/2011	2

CustomerName depends on CustomerID (non-key attribute)

2NF-> 3NF

- Action Required
  - Examine each attribute
  - If an attribute(s) does not depend on the whole key, or it depends on another non-key attribute, remove the attribute(s) and use attribute on which it depends on to form a new relation.
    i.e. create new table comprising the attribute(s) and the non-key attribute upon which it depends
  - Determine the key(s) for the new table(s)
  - Name the new table(s)

Data Illustration

Orders

OrderID[PK]	CustomerID	CustomerName	OrderDate	ProductsTotal
A1091	S009	Lynn Wang	21/7/2011	3
A1092	S010	Suzan Tan	12/1/2011	2
A1093	S010	Suzan Tan	21/8/2011	2

Orders

split table

Order ID [PK]	Customer ID	Order Date	Products Total
A1091	S009	21/7/2011	3
A1092	S010	12/1/2011	2
A1093	S010	21/8/2011	2

Customers

CustomerID	Customer Name
S009	Lynn Wang
S010	Suzan Tan

3NF

Orders
OrderID
CustomerID
OrderDate
ProductsTotal

Products
ProductID
ProductName
UnitPrice

OrderDetails
OrderID
ProductID
Qty

Customers
CustomerID
CustomerName

Normalization

ONF-> 1NF	• **Multivalue Attributes/Repeating Groups?** • Move multivalue/repeating group to a new entity -determine the key of the new relation
1NF ->2NF	• Are there **attributes dependent on a partial key** (of **composite key**)? •Move attribute(s) to a new entity -determine the key of the new entity
2NF ->3NF	• Any **non-key attribute dependent on any other non-key attribute?** • Move attribute(s) to a new entity -determine the key of the new entity
	Optimization - combining entities with same primary key / remove derivable attribute

Data Illustration

Orders

OrderID [PK]	CustomerID	OrderDate	Products Total
A1091	S009	21/7/2011	3
A1092	S010	12/1/2011	2
A1093	S010	21/8/2011	2

derivable

OrderDetails

Order ID [PK]	Product ID [PK]	Qty
A1091	S1001	100
A1091	S1003	200
A1091	S1005	250
A1092	S1001	10
A1092	S1004	50
A1093	S1001	80
A1093	S1005	90

3NF- Optimised

Orders
OrderID
CustomerID
OrderDate

Products
ProductID
ProductName
UnitPrice

OrderDetails
OrderID
ProductID
Qty

Customers
CustomerID
CustomerName

Optimisation -
Remove derivable attribute

Normalization : From 0NF to 3NF

0NF

Orders
OrderID
CustomerID
CustomerName
OrderDate
ProductID
ProductName
Qty
ProductsTotal

Orders
OrderID
CustomerID
CustomerName
OrderDate
ProductsTotal

OrderDetails
OrderID
ProductID
ProductName
Qty

Orders
OrderID
CustomerID
OrderDate
ProductsTotal

Customers
CustomerID
CustomerName

OrderDetails
OrderID
ProductID
Qty

OrderDetailsProduct
ProductID
ProductName

0NF->1NF 1NF->2NF 2NF->3NF

More on Optimization

- Optimization is combining tables that have identical primary keys

Table-1 (K1, K2, K3, DE1, DE2, DE3)

AND

Table-2 (K1, K3, K2, DE2, DE4, DE5)

ARE COMBINED TO GIVE

Table-3 (K1, K2, K3, DE1, DE2, DE3, DE4, DE5)

* where K is a Key-element and DE is a Data-element

- Key sequence is not important
- Don't lose data elements

More on Optimization

If we have:

Table-1 (K1, K2, DE1, DE2, DE3)

AND

Table-2 (K1, K2, K3, DE2, DE4, DE5)

CAN WE COMBINE?

More on Optimization

- Two approaches
  a. Optimize at the end of each normalization step
  b. Optimize after 3NF
- Recommend a combination of both

Benefits of Normalization

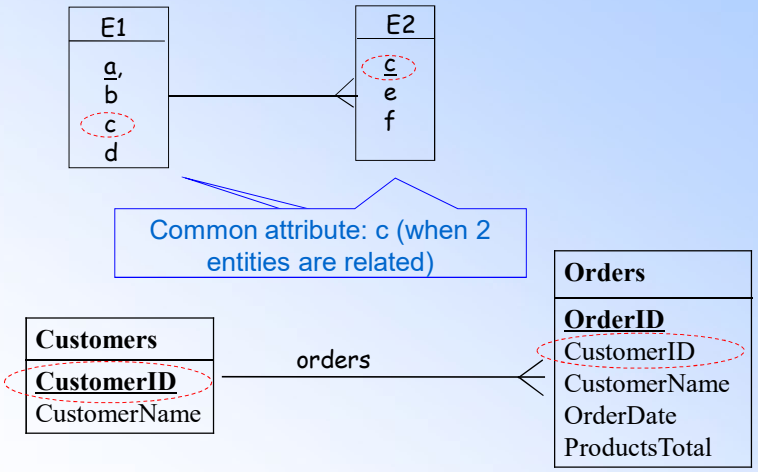
- Prevent data updates problem
- Reduce data redundancies
- Foundation for an optimum physical data base design using any DBMS

Data Analysis and Design

- Data Analysis
  - Entity Relationship diagram (data model)
  - Attribute Analysis
- Data Design
  - Normalization
  - Logical Data Model

Relationships derived from Normalization

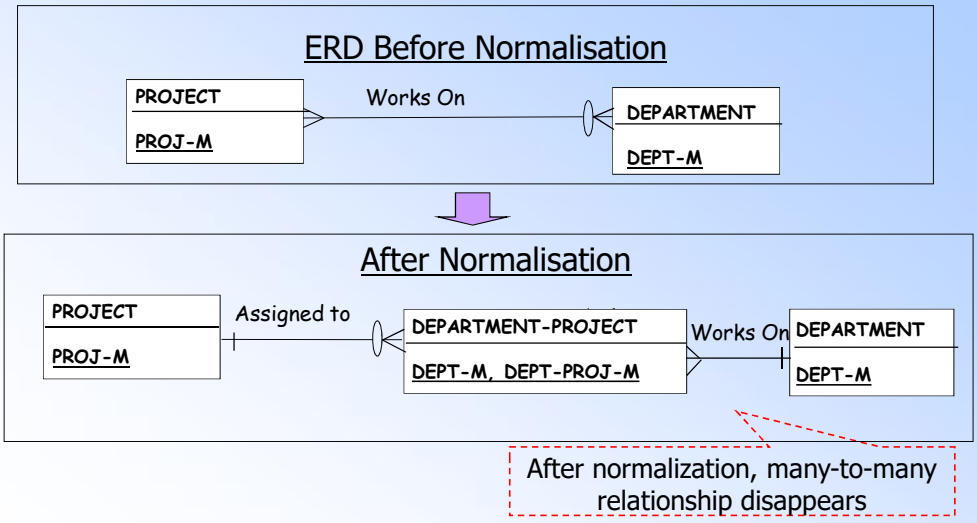
- Key Relationship



Logical Data Model

- The LDM is a more detailed representation of data which will be stored by the system. We can therefore, expect:
  - ❑Additional Entities (as a result) of normalization
  - ❑Additional Relationships (linking new and existing entities)

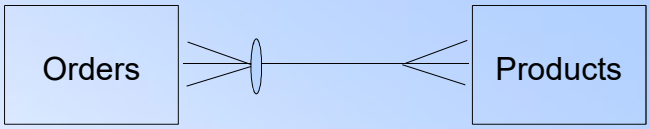
Relationships After Normalization – Department Project Example



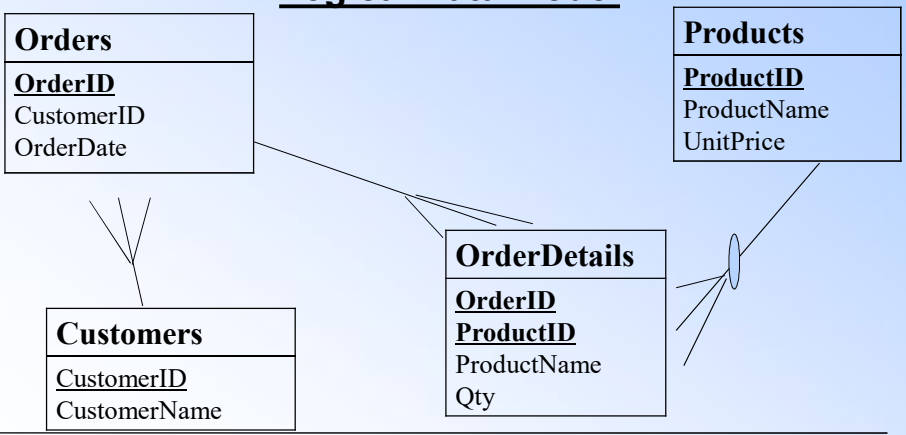
Developing Logical Data Model

- Establish the relations on 3NF tables
- Refer to:
  - ❑ Normalization Process
    - ❖If table B split from table A, establish the relationships between them
  - ❑ Original ERD

ERD (Before Normalisation)

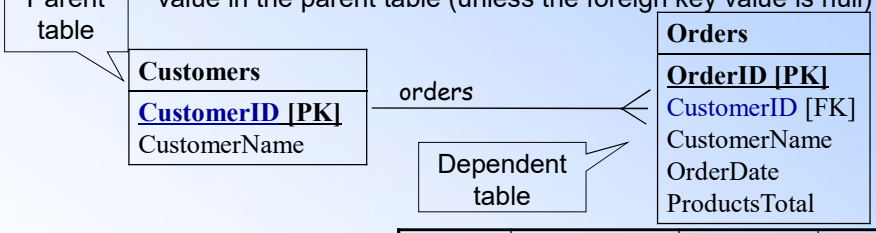


Logical Data Model



Keys Derived From Relationships

- **Foreign Keys** are identified
  - A foreign key enforces referential integrity
    - ↗ foreign key attribute must correspond to an existing primary key value in the parent table (unless the foreign key value is null)



CustomerID	Customer Name
S009	Lynn Wang
S010	Suzan Tan

Order ID [PK]	CustomerID	Order Date	Products Total
A1091	S009	21/7/2011	3
A1092	S010	12/1/2011	2
A1093	S010	21/8/2011	2

Relational Database

- Relational Database in the market:
  - Oracle
  - MySQL
  - SQLServer
  - … and many more

Mapping To Relationship Table

- Each Entity -> one relational table
  - each Attributes-> one column
  - identifier (key)-> primary key
  - foreign key -> foreign key

Summary

- Data Design
  - Normalization
    - 0NF
    - 1NF
    - 2NF
    - 3NF
  - Optimisation
  - Logical Data Model